View metadata, citation and similar papers at core.ac.uk
brought to you by CORE
provided by University of Queensland eSpace

Intermodal Path Algorithm for Time-Dependent Auto Network and Scheduled Transit Service

Alireza Khani, Sanggu Lee, Mark Hickman, Hyunsoo Noh, and Neema Nassir

A simple but efficient algorithm is proposed for finding the optimal path in an intermodal urban transportation network. The network is a general transportation network with multiple modes (auto, bus, rail, walk, etc.) divided into the two major categories of private and public, with proper transfer constraints. The goal was to find the optimal path according to the generalized cost, including private-side travel cost, public-side travel cost, and transfer cost. A detailed network model of transfers between modes was used to improve the accounting of travel times during these transfers. The intermodal path algorithm was a sequential application of specific cases of transit and auto shortest paths and resulted in the optimal intermodal path, with the optimal park-and-ride location for transferring from private to public modes. The computational complexity of the algorithm was shown to be a significant improvement over existing algorithms. The algorithm was applied to a real network within a dynamic traffic and transit assignment procedure and integrated with a sequential activity choice model.

In typical transportation networks, the dominant modes of transportation are auto and transit, and many studies have been done in each of these areas. Willingness to use public transportation has increased in recent decades because of economical and environmental issues. At the same time, many people prefer to access transit routes by car or bike, especially to access express bus routes and rail systems that may not be easily reached by walking. Thus, multimodal trips are commonplace, as faster transit modes have been developed in urban areas across the United States. However, there have been few studies on intermodal transportation problems. Modeling of intermodal trips and defining the optimal path using more than one mode, as well as finding the optimal park-and-ride location, are still open topics of discussion.

Users of a multimodal transportation network may either choose the mode of transportation first and then find the best path in the mode or compare the paths in different modes and choose the best of these (*1*). In the first approach, the chosen path is not necessarily optimal according to the defined generalized cost, whereas the latter approach gives the optimal path over all modes. However, in some cases a set of constraints on modes requires the user to choose the path within a subset of modes only. This happens in transportation modeling also, when an intermodal demand is generated in the previous step and a path in a constrained set of modes should be found and assigned to the passenger (*2, 3*).

This study approaches the problem of modeling an optimal path in an intermodal transportation network with some assumptions. Two categories of transportation are considered: private modes and public modes. Private modes are use of auto or a bicycle, and public modes are all kinds of bus routes, rail systems, and other means of urban transit. Properties for each category separate them by usage. In private modes, the user has the flexibility of departure time and of a variety of routes and corridors toward the destination, often with the ability to adjust the path en route on the basis of observed conditions on the roadway. In addition, at the destination (or at the transfer point from private to public mode), the private vehicle (car or bicycle) must be parked somewhere, which imposes parking cost, delay, and space constraints. A bicycle could be carried onto the transit vehicle, but available space on the vehicle remains as a constraint. On the other side, public modes are limited in temporal and geographical coverage, but the only concern in transferring between public modes is the inconvenience of a transfer. Walking is also a mode of transportation and can be included in both private and public modes.

In this study, an intermodal path is defined as a path that includes both private and public modes. This definition, which is rational and intuitive with traveler behavior, reduces the complexity of the problem, that is, the number of modes and mode changes. In this case, the important mode transfer in an intermodal trip is where the passenger changes from a private to a public mode (e.g., at park-and-ride facilities). Therefore, in the optimal path, the best location to transfer (i.e., the choice of where to park the car) is determined as well (*3*).

LITERATURE REVIEW

In one of the first studies in this area, Abdulaal and LeBlanc introduced the discussion on intermodal transportation modeling by presenting two ways to combine mode choice and route choice models (*4*). In one approach, mode choice and route choice are done sequentially; in the other, they are done simultaneously. In 1994, Fernandez et al. opened the discussion on trip planning with details on combined modes (*1*). Their study addressed two major issues in multimodal transportation modeling: (*a*) how a user chooses the mode of the trip, and then, depending on the answer to the first question, how the best route is chosen; and (*b*) how the transfer point

Department of Civil Engineering and Engineering Mechanics, University of Arizona, 1209 East 2nd Street, P.O. Box 210072, Tucson, AZ 85721-0072. Corresponding author: M. Hickman, mhickman@email.arizona.edu.

Transportation Research Record: Journal of the Transportation Research Board, No. 2284, Transportation Research Board of the National Academies, Washington, D.C., 2012, pp. 40–46.
DOI: 10.3141/2284-05

from the private to the public mode is selected. They proposed three approaches to modeling intermodal trips:

1. With the generalized cost of the combined mode, people choose their path on the basis of Wardrop's principle of optimality to minimize the cost of their trip.
2. The mode of the trip is determined by a mode choice model in which the combined mode is considered a pure mode, and then the shortest path is found in the selected mode.
3. An extension of the second model includes the choice of transfer point in the demand submodel.

Although Fernandez et al. proposed the third model as a new approach in the integration of demand models and network performance in an analytic framework (*1*), the network is assumed to be static.

Modesti and Sciomachen proposed an algorithm for finding a multiobjective shortest path in a multimodal transportation network (*5*). They introduced a utility function for weighting the links on the basis of their cost and time and then used the classical Dijkstra shortest path algorithm to find the path with maximum utility. Ziliaskopoulos and Wardell developed an algorithm for finding the intermodal least time paths on multimodal networks with time-dependent link travel times and turning delays (*6*). Their label-correcting algorithm was designed for all time intervals, and its complexity is $O(T^2V^5)$, independent of the number of modes, where T is number of time intervals and V is the number of nodes. Their computational experiments showed that their algorithm has a practical computational time that is linear with the number of nodes and time intervals. Lozano and Storchi also applied a label-correcting algorithm to find the shortest viable hyperpath with a predefined maximum number of modal transfers (*7*). The approach is useful when there is no exact schedule for the transit system (i.e., the transit network is frequency based). Because it considers more than one criterion, the algorithm's result is not necessarily optimal, and users can choose the best hyperpaths from the output, according to their preferences.

A multimodal assignment formulation was proposed by Garcia and Marin in the form of a variational inequality considering the combined modes (*3*). For the equilibrium model, they used a nested logit model that captured the choice of mode and transfer point between modes as well as routes. Garcia and Marin formulated the problem in a hyperpath space and performed stochastic assignment with elastic demand (*3*). Zhou et al. developed an integrated framework to model choices of departure time, mode, and path in a multimodal transportation system (*8*). As a part of the model, a time-dependent least-cost path algorithm based on the work of Ziliaskopoulos and Wardell is used to generate intermodal paths (*6*). For this algorithm, a set of constraints for possible mode transfer is applied.

A review of the most important studies in this area suggests a more efficient and flexible algorithm for intermodal path generation. This study presents an algorithm with less computational complexity that can be easily implemented in a multimodal network.

METHODOLOGY

A path is divided into a private mode, a public mode, and a connection between them. For the private mode, a multisource time-dependent shortest path (MTDSP) is considered. For the transit side, which usually is more complex and for which finding the best path requires more computational time, a trip-based, time-dependent shortest path algorithm (TBSP) is used that takes advantage of the hierarchical structure of transit systems. Geographic information system (GIS) layers, including auto, walking, and transit networks, are used in the investigation of the connection between auto and transit networks to estimate the time required to transfer from auto to transit within each park-and-ride facility.

Modeling Intermodal Transportation Network at Park-and-Ride Facilities

For the description of an intermodal transportation network, the complete transport chain (e.g., a set of intermodal paths) should be mapped adequately in the model, especially for the transfer between modes (Figure 1). The proposed method moves toward a more realistic behavioral representation of the traveler's path in the vicinity of each park-and-ride and considers the accessibility and movement of travelers and vehicles. Although the output shows simply the travel times of a set of intermodal paths, this method explicitly captures the interaction between the auto and the transit network, that is, captures access to park-and-ride by auto, parking the car, and walking to the transit stop.

To facilitate this interaction, aerial-photo-based access points are introduced for each park-and-ride. These access points represent driveways from (to) the road network to (from) the park-and-ride lot. Each access point can be placed in the auto network as a node along the road network. The Euclidean distance from or to a street junction (e.g., intersection or nearest node) also can be easily applied.

To model the transfer explicitly, all the access points in the auto network that are on the perimeter of the park-and-ride lot should be connected to all the transit stops within the park-and-ride facility. This is achieved by connecting the access point to the centroid of the park-and-ride lot with an auto link, to complete the auto part of the mode transfer. Then, the addition of walk links from the centroid of the park-and-ride to the transit stops or stations in the transit network can establish an intermodal transportation network.

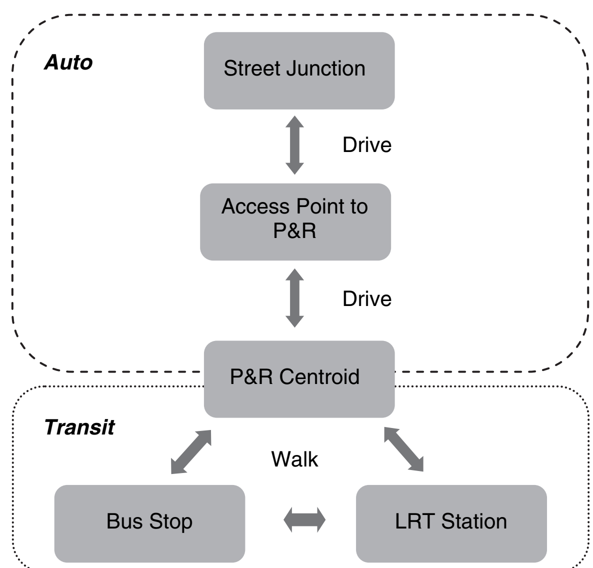

FIGURE 1 Framework of intermodal network at park-and-ride facilities (P&R = park-and-ride; LRT = light rail transit).

In the same way, the set of intermodal paths can be enumerated to estimate travel times between each stage. These paths enhance the representation of travel times (costs) and transfer behaviors in the vicinity of a park-and-ride and can be more practical in modeling intermodal travel. Figure 2 shows an intermodal transportation network that consists of various modal networks. Within these networks, a traveler can move from an access point in the auto network to an explicit transit stop location in the transit network.

This disaggregate approach of developing intermodal paths can be useful for incorporating any additional information (e.g., congestion) into the associated stage. For example, the generalized cost of the mode transfer links may include a penalty representing the uncertainty of finding a parking spot (*9*). Transfer cost (from auto to transit) may include in-vehicle time from the access point to an available parking space, walking between modes, and waiting time for the transit vehicle. For example, in the presented study, speeds in the park-and-ride lot are modeled as 10 mph.

In addition to the typical values for each part of the transfer, additional delay may be considered. The main sources of this additional delay are vehicle deceleration before accessing an access point and time required for entering the lot (e.g., taking a ticket or paying a fee), finding a parking space (in congested parking lots), and parking the car.

TBSP Algorithm

There are many algorithms for finding the shortest or the optimal path in a transit network. Some of these algorithms are for transportation planning and give one-to-all paths, and some are for itinerary planning and give a one-to-one path in the output. The TBSP algorithm takes advantage of the transit schedule, in the form of detailed vehicle scheduled trips, and of the hierarchical property of transfer stops to find the shortest path more quickly than typical labeling algorithms can.

Use of public transit data sources such as Google's general transit feed specification (GTFS) allows the flexibility needed to develop efficient models and algorithms for public transportation research (*10*). In the TBSP, when a node in a scan-eligible list is processed, the labels of all the stops along a transit vehicle trip are updated in one iteration. That is, starting from current stop *s* and taking transit trip *tr,* instead of updating only the label of the adjacent stop all labels for the stops in the trip are updated at once. In the case in which the passenger's origin and destination are connected via the same route, the direct path (which is often the shortest and the optimal path because it has no transfer) is found quickly. Otherwise, the passenger, after getting on the vehicle, does not consider every stop as a possible alighting point but instead gets off at a real transfer stop. For example, a midblock stop with one passing route is not a transfer stop because it neither is served by another route nor is within short walking distance of another stop. Therefore, transfer stops are determined at a higher level of the transit network (i.e., through use of the hierarchical structure of the transit network) and uses as the scan-eligible stops. That is, when the label of a stop is updated, the stop will be added to the list of stops to be scanned later only if it is a transfer stop. This technique significantly reduces the number of iterations in the algorithm if only a few stops are true transfer stops in the transit network. In a typical schedule-based transit shortest path (*11*), the complexity of the algorithm is $O(S^2)$, where S is the number of stops. In the TBSP, because only the transfer stops are added to the scan-eligible list, the complexity of the algorithm is $O(S'^2)$ when S' is the number of transfer stops.

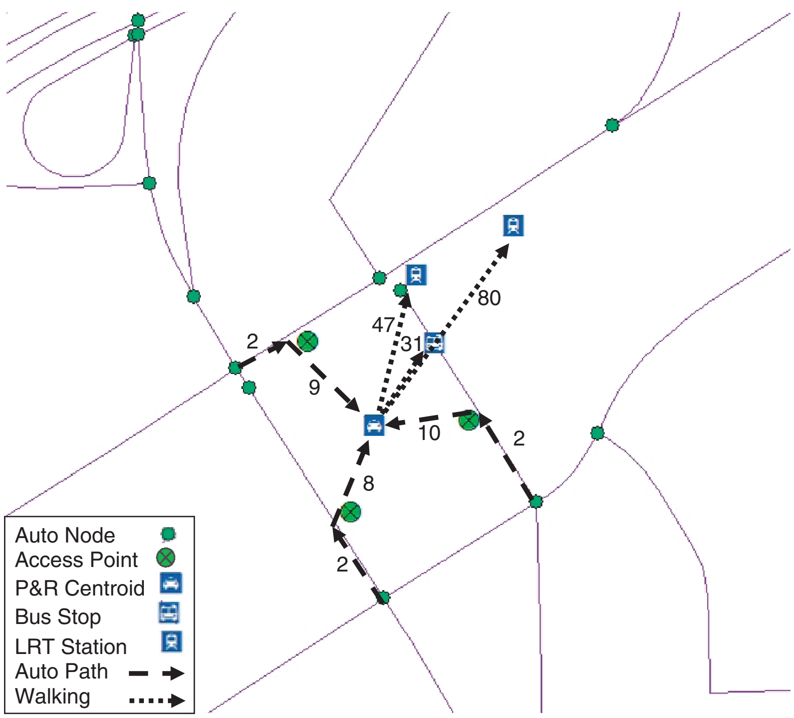

FIGURE 2 Intermodal network, including layers of transportation modes, connection between nodes and transit stops, and time of each part in seconds at Sunrise park-and-ride, Rancho Cordova, California.

Intermodal Algorithm for Optimal Path

In a multimodal transportation network with P park-and-ride facilities, choosing the optimal point to transfer from auto to transit is similar to a multidestination choice problem, and the complexity of the transfer problem increases linearly with the number of choices. The problem is defined as finding the optimal path between an origin and a destination with a preferred arrival time (PAT) at the destination with all available transportation modes considered (Figure 3). If the park-and-ride points are defined as the alternatives for an auxiliary destination and the final destination is called the main destination, the model determines the best choice among the alternatives for the auxiliary destinations considering the overall trip to the main destination.

The structure of the model, including files and subroutines, is shown in Figure 4. The main inputs are GTFS data for a transit network, an auto network including node and link geometry and properties, time-dependent link travel times in the auto network, aerial photos of the park-and-ride facilities, and the detailed roadway network in GIS format. In addition, the park-and-ride model is used to generate access and egress links between nodes in the auto network and the transit stops, as well as mode change links at park-and-ride locations, for use in the algorithm.

To meet the PAT at the destination, and to ensure the use of transit, a backward TBSP is run from the destination beginning at the PAT, and all the stops are labeled if there is available service for reaching the destination by the PAT. The main file used in this subroutine is the transit schedule in GTFS format. First, the main destination node is selected, and the travel times on the access links are used to label the transit stops within walking distance from it. Then the TBSP is run from these labeled stops to find the path from all other stops in the network. After this step, potential access to all transit stops from the auto network (i.e., locations for an intermodal transfer) is established. At this point, the label of each node (park-and-ride lot) is the travel cost from that node to the destination via transit only.

The connection between the auto and the transit network is then established by mode change (park-and-ride) coding described in the previous section. This means that from the transit stops in the park-and-rides (i.e., the auxiliary destinations), the transfer cost is used to label the neighboring nodes in the auto network.

On the auto side, instead of finding the shortest path from each park-and-ride facility to the origin separately, an MTDSP algorithm is used; this label-correcting algorithm uses more than one node as the initial source in the algorithm. The nodes in the auto network that are updated from the park-and-ride facilities keep their labels and are added to the scan-eligible list. The MTDSP finds the best transfer location (i.e., park-and-ride) from the origin, considering the cost for the transit part of the trip. The initialization step of MTDSP is as follows:

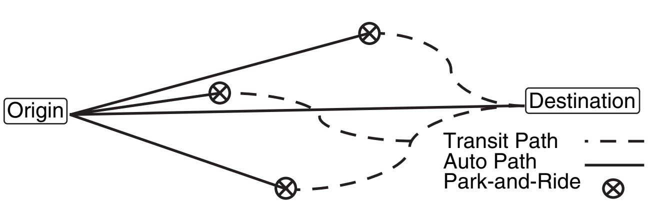

FIGURE 3 Intermodal trip through park-and-ride facility.

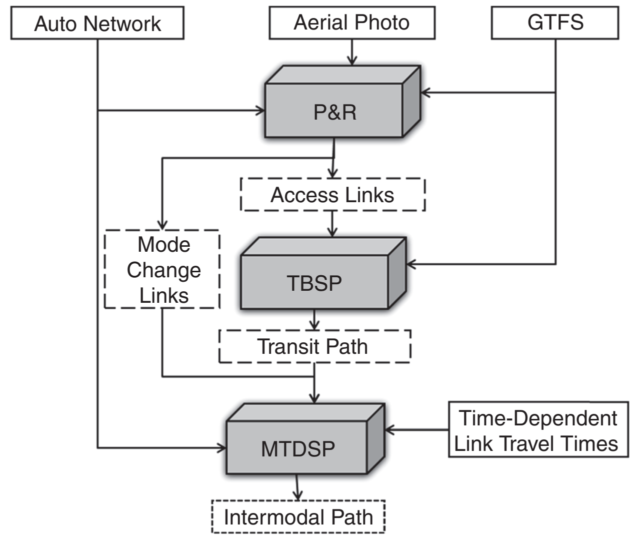

FIGURE 4 Structure of intermodal optimal path algorithm.

1. Add the destination with label zero to the scan-eligible list.
2. Set the labels of all the nodes to infinity and the predecessor nodes to null, except those nodes that have a label and a predecessor from the execution of TBSP in the transit network.
3. Add nodes updated from mode change links (access point nodes from the park-and-ride lot) to the scan-eligible list.

This technique of adding multiple nodes to the scan-eligible list at the beginning of the algorithm, introduced by Festa (*12*), offers the advantage of finding the best source (park-and-ride) node to the target (the origin) by automatically comparing the travel cost from the destination to each source node. Because the initial labels of the source nodes are set by the TBSP algorithm, the MTDSP also takes into account the transit travel cost to choose the best park-and-ride location overall. In the conventional approach, a TDSP must be calculated from each source node to the origin, which can be a time-consuming procedure when there are many park-and-ride facilities in the network.

For the implementation of the algorithm, the auto network is defined by a graph $G_A(N, L)$, where N is the set of nodes, indexed by n, and L is the set of links, indexed by l. The transit network is defined by a graph $G_T(S, R)$, where S is the set of stops, indexed by s, and R is the set of routes, indexed by r. Each route contains a set TR that is the set of trips tr. Each trip also contains a subset of stops to serve, as well as the schedule time for each stop. There is also a set of transfer links $tf(s_1, s_2, w_{tf}) \in TF$ that includes the walking time between a pair of transfer stops, indicated by w_{tf}. The access links $a(n, s, t_a) \in A$ connect a node n and a stop s with the walking time t_a. The mode change links $m(n, s, t_m) \in M$ also have the same format as the access links, but the stops are the source nodes. The input of the algorithm is the passenger's origin, $O \in N$, destination, $D \in N$, and PAT to the destination. Finally, the shortest intermodal path algorithm is as follows.

Step 0. Initialization.

- Set Label(n) = ∞, Pre(n) = null, and Mode(n) = null for $n \in N$.
- Set Label(s) = ∞, Pre(s) = null, Trip(S) = null, and Mode(n) = null for $s \in S$.
- Get O, D, and PAT, and set Label(D) = PAT.

Step 1. Egress to the destination from transit by walking.

- For all $a \in A$ with $n_a = D$: Label(s_a) = PAT − t_a, Pre(s_a) = D, Trip(s_a) = W, Mode(s_a) = T, and add s_a to the scan-eligible list.

Step 2. Transit shortest path.

- Run the TBSP backward from nodes in the scan-eligible list to the source nodes. In this step, for any $s \in S$, if Label(s) is updated, set Mode(s) = T.

Step 3. Access to transit by walking.

- For all $a \in A$, if Label(s_a) ≠ ∞: Label(n_a) = Label(s_a) − t_a, Pre(n_a) = s_a, and Mode(n_a) = T.

Step 4. Access to the destination by auto.

- Add the destination node D with Label(D) = 0 and Pre (D) = Null to the scan-eligible list.

Step 5. Access to transit by auto.

- For all $m \in M$, if Label(s_m) ≠ ∞: Label(n_a) = Label(s_a) − t_m, Pre(n_a) = s_a, and Mode(n_a) = T, and add n_a to the scan-eligible list.

Step 6. Auto shortest path.

- Run the MTDSP backward from the nodes in the scan-eligible list to the origin. In this step, for any $n \in N$, if Label(n) is updated, set Mode(n) = A.

Step 7. Backtrack the path.

- Trace the path starting from the origin O by using Pre(n) and Mode(n).

In the case that an optimal intermodal path (rather than a shortest path) is required, a label for the generalized cost is used and a weighting system (f_W, f_A, f_T, f_M, p_m) is applied to the times in each part of the path. The parameters are as follows:

- f_A, weight of travel time by auto, used in MTDSP;
- f_T, weight of travel time by transit, used in TBSP;
- f_W, weight of walking time, applied to access links and transfer links;
- f_M, weight of mode transfer times, used in Step 4; and
- p_m, additional penalty cost for the mode transfer.

The complexity of the full algorithm is $O(S^2 + P + N^2)$, where S is the number of stops, P is the number of park-and-ride locations (source nodes), and N is the number of nodes in the auto network. In general, if S and N are comparable, the complexity of the algorithm is dominated by the transit side. In the typical schedule-based transit shortest path algorithm, the complexity is $O(S^2)$, but in the TBSP, the number of iterations is decreased by taking advantage of the transfer stop hierarchy. Therefore, the complexity is $O(S'^2)$, where S' is the number of transfer stops in the network. For finding all-to-all shortest intermodal paths in all time intervals T, the whole computational effort will be $O(\text{TNS}'^2)$, which is better than existing intermodal path methods in the literature (*6*).

Since the complexity of the algorithm is not related to the number of park-and-ride facilities, the procedure can be used for modeling kiss-and-ride trips. In kiss-and-ride trips, the passenger is dropped off at a transit stop from a shared ride, and no car must be parked. That is, in kiss-and-ride trips, access to transit can be made at virtually any stop. Yet these trips can be modeled properly by the proposed algorithm. For the proposed algorithm to model kiss-and-ride trips, a modification was made to Step 3. The modified Step 3 connects the transit and auto networks with the access links as follows:

For all a $\in A$, if Label(s_a) ≠ ∞: Label(n_a) = Label(s_a) − t_a,
Pre(n_a) = s_a, and Mode(n_a) = T and add n_a to the
scan-eligible list.

The proposed approach compares all the paths in all combination of modes for the optimal path and does not guarantee that both auto and transit modes are used. This outcome is intuitively rational, because it results in a single-mode path in the extreme cases in which either auto-only or transit-only paths dominate the multimodal paths. The examples are the case in which the origin is directly connected to the destination by a transit route (e.g., they are relatively close), and the use of an auto is not required to access transit stops, and, at the other extreme, the case in which there is no attractive transit service or park-and-ride location on the way from the origin to the destination, and commuting by auto is faster or more cost-effective. Finally, a slight modification of the algorithm can be used to model the cases in which a multimodal path is required, but this may not necessarily result in the optimal intermodal path.

MODEL FOR INTEGRATED DYNAMIC TRAFFIC AND TRANSIT ASSIGNMENT WITH ACTIVITY CHOICE

Because the proposed algorithm gives the intermodal shortest path from all origins to a destination, and the required computational time is on the order of a single shortest path, it is suitable for planning applications. This study was motivated in part by modeling the intermodal trips in a real project that integrates an activity-based model and a dynamic traffic and transit assignment.

The transportation group of the University of Arizona is developing the dynamic traffic and transit assignment and has developed a traffic simulation model called MALTA (multiresolution assignment and loading of transportation activities) and a transit assignment and simulation tool called FAST-TrIPs (flexible assignment and simulation tool for transit and intermodal passengers), which integrates with MALTA. The travel demand generated by an activity-based model, including auto, transit, and intermodal trips, is simulated by MALTA and FAST-TrIPs, and the results, including the experienced travel times, are reported to the activity-based model as feedback. The procedure continues through several iterations to reach equilibrium. Assigning a path to the trips requires a suitable shortest or optimal path algorithm. Figure 5 shows the overall procedure of the integrated model.

The model was tested in the real network of Rancho Cordova, California (Figure 6). This network contains 447 nodes and 850 links in the auto network and includes a local transit service that is part of the Sacramento Regional Transit system. The transit network in Rancho Cordova contains five bus routes and a light rail line and covers 163 stops. There are two park-and-ride facilities in the network. On a typical weekday, more than 70,000 trips are generated in the area during the morning peak (6:00 to 9:00 a.m.). Although this network is relatively small and many of the intermodal trips are headed to downtown Sacramento, California, it is an appropriate network for testing purposes.

The proposed intermodal shortest path algorithm is coded in C++ and is tested with a typical personal computer (Core 2 Duo, 2.50 MHz CPU and 2 GB RAM). The results show that the algorithm has good performance. All the intermodal paths in the network, for approximately 200,000 origin–destination pairs and a single time interval in the morning peak (i.e., PAT = 8:00 a.m.), are generated in about 38 s computation time. However, many of the optimal paths generated use only one mode of transportation. This result was expected in this relatively small network.

To more closely test the performance of the algorithm, destinations were selected that are served by an attractive transit service and for which an intermodal path is likely to be found. Then

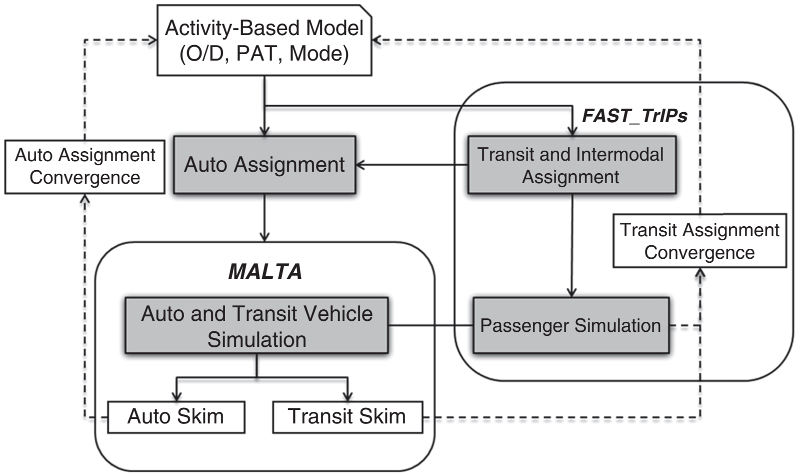

FIGURE 5 Structure of integrated model in urban continuum (O/D = origin–destination).

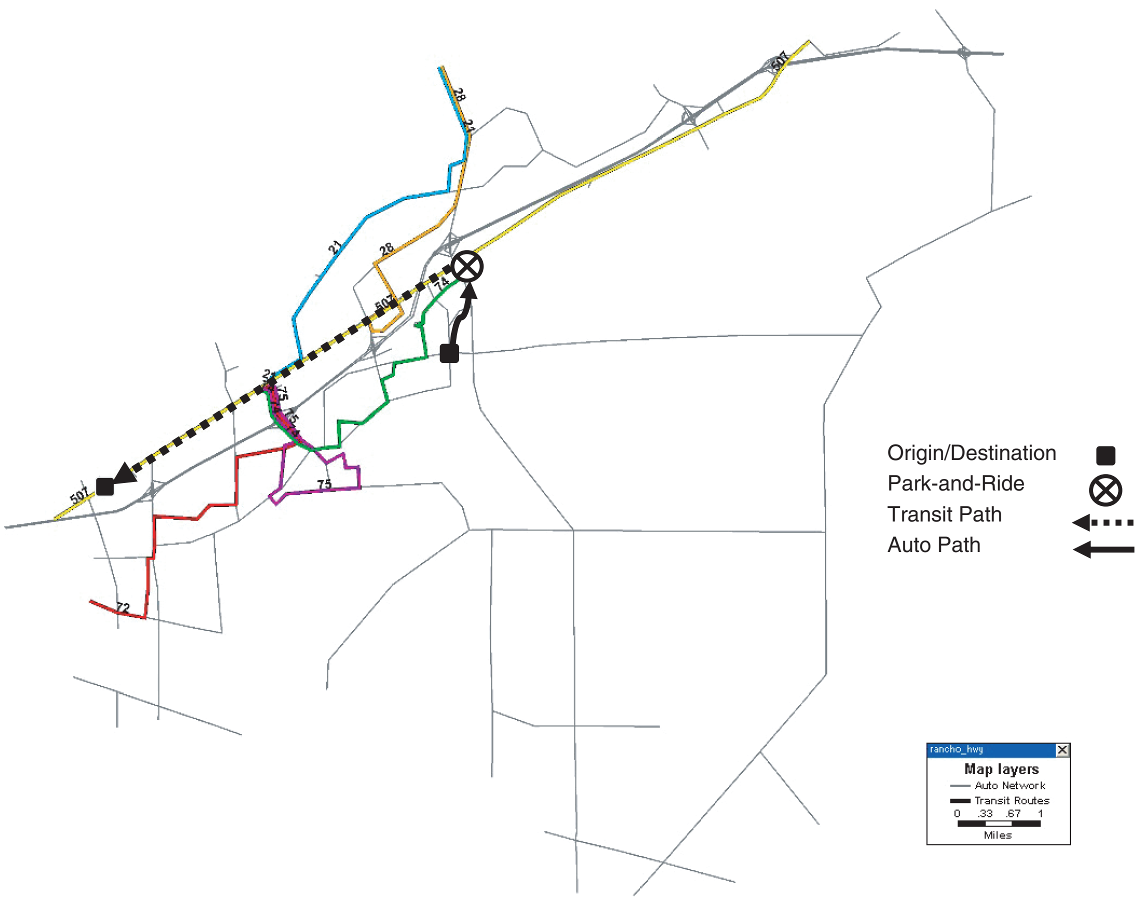

FIGURE 6 Example on Rancho Cordova intermodal network.

TABLE 1 Results of Case Study and Improvements in Computational Time

Algorithm	Time per Destination (s)	Improvement (%)
Baseline case	0.85	—
With TBSP but not MTDSP	0.80	6
With MTDSP but not TBSP	0.26	69
With both TBSP and MTDSP (proposed algorithm)	0.21	75

NOTE: — = not applicable.

the algorithm was run for four cases. The results are provided in Table 1. The baseline case, the easiest but most time-consuming, is to run a transit shortest path backward from the destination and to run an auto shortest path from each park-and-ride location to the origin; the results are then compared to find the best path. The second case is the same as the baseline case, except that the TBSP is used for the transit shortest path. In the third case, the MTDSP is used instead of running several auto shortest paths, but for the transit network, a schedule-based transit shortest path is used (not TBSP). The fourth case is the proposed algorithm in this study, which utilizes TBSP and MTDSP. Results indicate that the component algorithms have a significant effect on the computational performance. In networks with many park-and-rides, the improvements in computation time are expected to be much greater, because the complexity of the algorithm is not related to the number of park-and-ride lots.

In Figure 6, an example of an intermodal path is shown from Origin Node 280 to Destination Node 35 with a PAT equal to 8:00 a.m. Although there is no transit coverage in the vicinity of the origin node, an auto takes 26 min to reach the destination. The travel time for the shortest intermodal path for the sample origin–destination pair is 23 min, including 4 min for driving to the Sunrise park-and-ride, 2 min for the mode transfer to transit, 15 min for travel by light rail transit, and 2 min to walk to the destination.

CONCLUSION

In this study, an intermodal optimal path algorithm was developed that takes advantage of a trip-based transit shortest path algorithm and a MTDSP algorithm for the auto segment of the trip. The complexity analyses show that the proposed algorithm is comparable to a single transit shortest path algorithm and gives the all-to-one intermodal path tree. Although the complexity of the algorithm is independent of the number of park-and-rides, it can also be applied to kiss-and-ride trips, in which finding the best transfer point is more complex. For the connection between the auto and the transit network, a detailed analysis was performed that models the park-and-ride facilities by using different sources of data.

The algorithm was tested in a real network, and results show that it has good performance and can be used in planning applications. In addition, multisource implementation of the shortest path can be used to solve the destination choice problem, in which there may be several options for the destination of a passenger trip. With no additional computational burden, the algorithm gives the best choice for the park-and-ride facility (the auxiliary destination), as well as the best path from the origin to the destination. As an extension of the model, the optimal intermodal tour can be investigated when the intermodal path should consider the return trip or trips to the other activity locations. The authors have also studied the latter topic (*13*).

ACKNOWLEDGMENTS

This study was conducted by the members of the University of Arizona Transit Research Unit. The authors appreciate the ideas and support of the unit's members.

REFERENCES

1. Fernandez, E., J. de Cea, M. Florian, and E. Cabrera. Network Equilibrium Models with Combined Modes. *Transportation Science,* Vol. 28, No. 3, 1994, pp. 182–192.
2. Friedrich, M. A Multimodal Transport Model for Integrated Planning. Presented at 8th World Conference on Transport Research. 1998. http://www.ptvamerica.com/fileadmin/files_ptvamerica.com/library/1998%20WCTR%20Conf%20VISUM%20multi-modal%20planning.pdf.
3. Garcia, R., and A. Marin. Network Equilibrium with Combined Modes: Models and Solution Algorithms. *Transportation Research Part B,* Vol. 39, No. 3, 2005, pp. 223–254.
4. Abdulaal, M., and L. LeBlanc. Methods for Combining Modal Split and Equilibrium Assignment Models. *Transportation Science,* Vol. 13, No. 4, 1979, pp. 292–314.
5. Modesti, P., and A. Sciomachen. A Utility Measure for Finding Multiobjective Shortest Paths in Urban Multimodal Transportation Networks. *European Journal of Operational Research,* Vol. 111, 1998, pp. 495–508.
6. Ziliaskopoulos, A., and W. Wardell. An Intermodal Optimum Path Algorithm for Multimodal Networks with Dynamic Arc Travel Times and Switching Delays. *European Journal of Operational Research,* Vol. 125, No. 3, 2000, pp. 486–502.
7. Lozano, A., and G. Storchi. Shortest Viable Hyperpath in Multimodal Networks. *Transportation Research Part B,* Vol. 36, No. 10, 2002, pp. 853–874.
8. Zhou, X., H. Mahmassani, and K. Zhang. Dynamic Micro-Assignment Modeling Approach for Integrated Multimodal Urban Corridor Management. *Transportation Research Part C,* Vol. 16, No. 2, 2008, pp. 167–186.
9. Hull, E. Application of a Park-and-Ride Forecasting Procedure in the Greater Vancouver Transportation Model. Presented at 13th Annual International EMME/2 Users' Group Conference. 1998. http://www.inro.ca/en/pres_pap/international/ieug98/msword4.pdf.
10. General Transit Feed Specification Reference. http://code.google.com/transit/spec/transit_feed_specification.html. Accessed July 2011.
11. Tong, C. O., and A. J. Richardson. A Computer Model for Finding the Time-Dependent Minimum Path in a Transit System with Fixed Schedule. *Journal of Advanced Transportation,* Vol. 18, No. 2, 1984, pp. 145–161.
12. Festa, P. The Shortest Path Tour Problem: Problem Definition, Modeling, and Optimization. *Proc., International Network Optimization Conference,* Pisa, Italy, European Network Optimization Group, 2009.
13. Nassir, N., A. Khani, M. Hickman, and H. Noh. Algorithm for Intermodal Optimal Multidestination Tour with Dynamic Travel Times. In *Transportation Research Record: Journal of the Transportation Research Board, No. 2283*, Transportation Research Board of the National Academies, Washington D.C., 2012, pp. 57–66.

The Transportation Network Modeling Committee peer-reviewed this paper.